CLAIMS

1. Load balancing apparatus for a data communications network, the apparatus comprising:

hash logic for computing a hash function on incoming data packets;

a threshold detector connected to the hash logic for triggering, in response to utilization of the downstream objects exceeding a predefined threshold, redefinition in the hash logic of parameters of the hash function from a first set of parameters to a second set of parameters for redistributing the data packets amongst the downstream objects; wherein,

the hash logic, in use, directs the packets for routing to downstream objects in the network via a first routing path based on a hash computation using the first set of parameters, and, if the threshold is exceeded, for selectively directing the packets to one of the first routing path and a second routing path in dependence on separate hash computations using the first and the second sets of parameters for subsequent routing of the packets via the selected one of the first and second routing paths based on the results of one of the separate hash computations.

2. Apparatus as claimed in claim 1, wherein the hash logic in use directs the data packet to the first routing path if the results of the separate hash computations intersect and otherwise directs the data packet to the second routing path.

3. Apparatus as claimed in claim 1 or claim 2, further comprising a filter connected to the hash logic for selectively bypassing the hash logic for flows having a lifetime exceeding a predefined value.

4. Apparatus as claimed in claim 4, further comprising the first routing path and the second routing path, the first routing path comprising first routing logic connected to the hash logic, and the second routing path comprising second routing logic connected to the hash logic, wherein the first routing path is faster than the second routing path, and wherein, on the second routing path, downstream objects are selected based on packet flow status.

5. Apparatus as claimed in claim 4, wherein the first routing logic comprises at least one network processor and the second routing logic comprises at least one general purpose processor.

6. Apparatus as claimed in claim 4 or claim 5, wherein the second routing logic is configured to detect a flow delimiter in a flow of data packets and, on detection of the start indictor, to route the corresponding flow according to the hash function using the second parameters.

7. Apparatus as claimed in claim 6, wherein the second routing logic is configured to detect flows of packets exceeding a predetermined inactivity time and to route such flows according to the hash function using the second parameters.

8. Apparatus as claimed in claim 7, wherein the second routing logic is configured to detect flows of packets exceeding a predetermined lifetime and to direct such flows to the first routing logic.

9. An application specific integrated circuit comprising load balancing apparatus as claimed in any preceding claim.

10. A network infrastructure node comprising load balancing apparatus as claimed in any of claims 1 to 7.

11. A data communications network comprising a network infrastructure node as claimed in claim 9.

12. A method of load balancing in a data communications network, the method comprising:

computing a hash function on incoming data packets;

triggering, in response to utilization of the downstream objects exceeding a predefined threshold, redefinition of parameters of the hash function from a first set of parameters to a second set of parameters for redistributing the data packets amongst the downstream objects; and,

directing the packets for routing to downstream objects in the network via a first routing path based on a hash computation using the first set of parameters, and, if the threshold is exceeded, selectively directing the packets to one of the first routing path and a second routing path in dependence on separate hash computations using the first and the second sets of parameters for subsequent routing of the packets via the selected one of the first and second routing paths based on the results of one of the separate hash computations.

13. A method as claimed in claim 11, comprising directing the data packets to the first routing path if the separate hash computations intersect and otherwise directing the data packets to the second routing path.